REMARKS

Claims 1-18 are pending in the present application. Claims 1-3, 5, 7-9, 11, 13-15 and 17 were amended. Reconsideration of the claims is respectfully requested.

Applicants' attorney and the Examiner discussed the application by phone on March 7, 2005. Applicants, through their attorney, express appreciation to the Examiner for granting this interview and for his helpful comments. A Statement of Substance of the Interview provided by Applicants is enclosed herewith.

I. 35 U.S.C. § 112, Second Paragraph

The Examiner has rejected Claims 5-6, 11-12 and 17-18 under 35 U.S.C. § 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which Applicants regard as their invention. In response to this rejection, Applicants have amended Claims 5, 11 and 17 to recite certain elements, pertaining to reverse proxy servers, with enhanced clarity and definiteness.

In view of this amendment, the rejection of Claims 5-6, 11-12 and 17-18 under 35 U.S.C. § 112, second paragraph has been overcome.

II. 35 U.S.C. § 103, Obviousness

The Examiner has rejected claims 1-18 under 35 U.S.C. § 103 as being unpatentably obvious, in view of U.S. Patent No. 6,438,592, to Killian, combined with U.S. Patent No. 6,185,598, to Farber et al. This rejection is respectfully traversed.

III. Response to Rejection of Claims 1, 7 and 13

In making their invention, Applicants were concerned with Quality of Service (QoS) in regard to the time required for servers in a network to respond to requests for electronic documents. Applicants were particularly concerned with document retrieval time in a distributed network environment. Applicants recognized that in such an environment, an original request can bring back a container page as a response, wherein the container page includes one or more inline elements. New requests must then be generated, in order to retrieve the respective inline elements. Applicants, in their invention, sought to provide an improved QoS that could readily determine the total time required to complete retrieval for both the parent request and all its inline content, collectively. Applicants recognized further that different inline elements may have to be delivered by different servers.

These purposes and concerns of Applicants are set forth in their specification such as at page 10, line 25 through page 11, line 5, and page 11, lines 17-20:

> The prior art QoS technology can provide individual times on each element as long as all of the requests are obtained from the same web server. However, this is not the typical case for retrieving a web page. In a distributed environment, it is not reasonable to assume that the web server that responded to the request for the container page is the same web server that will be asked to deliver all, or even any, of the inline elements of the page. Even if the QoS that serviced the original request (container page) received a new request for one of the inline members of the container page, it has no mechanism for recognizing the new request as a member of the original request. (Emphasis added.)
>
> The goal is for QoS to be able to recognize all of the members of the request and produce a metric that states "the total time for the parent request and all of its inline content to be save was 'x'".

Claim 1 as now amended is directed to a method for service time analysis in a computer network. The method of Claim 1 generally includes the steps of receiving a request from a network client for an electronic document; forwarding the request to an origin server and receiving a response stream that is sent to the client, the response stream containing an electronic document; and receiving at least one URI request from the client, wherein each URI request is for a resource embedded within the electronic document. To achieve Applicants' purposes as described above, Claim 1 further recites the following Features:

> Feature (1) Creating a data structure, such as a table, of key/service time value pairs, wherein each pair is associated with one of the URI requests, and each key is an identifier present in its associated URI request.

Feature (2) Forwarding each URI request to the origin server, and receiving a corresponding URI response from the origin server.

Feature (3) Updating each of the service time values in the data structure to indicate the time value for completing the response to each of the URI requests.

Feature (4) Sending each URI response to the client machine.

Feature (5) Using the keys and updated service time values from the data structure to compute the total time required to respond to all the requests associated with the electronic document, including all of the URI requests.

The above claim Feature (1) is disclosed in the application such as at page 10, lines 17-24. Feature (3) is taught at page 12, lines 18-22 and also at page 16, lines 18-20. Feature (5) is taught at page 12, lines 25-30 and page 13, lines 1-3.

It is to be understood that the data structure of key/time value pairs, now recited in the above Features (1), (3) and (5) of Applicants' Claim 1, is an essential element for achieving Applicants' goals and purposes. Respective key/time value pairs are used to track or monitor the request for and retrieval of each inline element in a container page, even when different web servers must be used to deliver different inline elements. After all the responses to respective requests for inline elements are completed, the updated time values in the data structure are available to compute the total service time, from initial request, required to retrieve both the container page and all the inline elements.

Applicants consider that Claim 1 patentably distinguishes over the Killian and Farber et al references, and any combination thereof, particularly in reciting the above Features (1), (3) and (5) in the over-all combination of Claim 1.

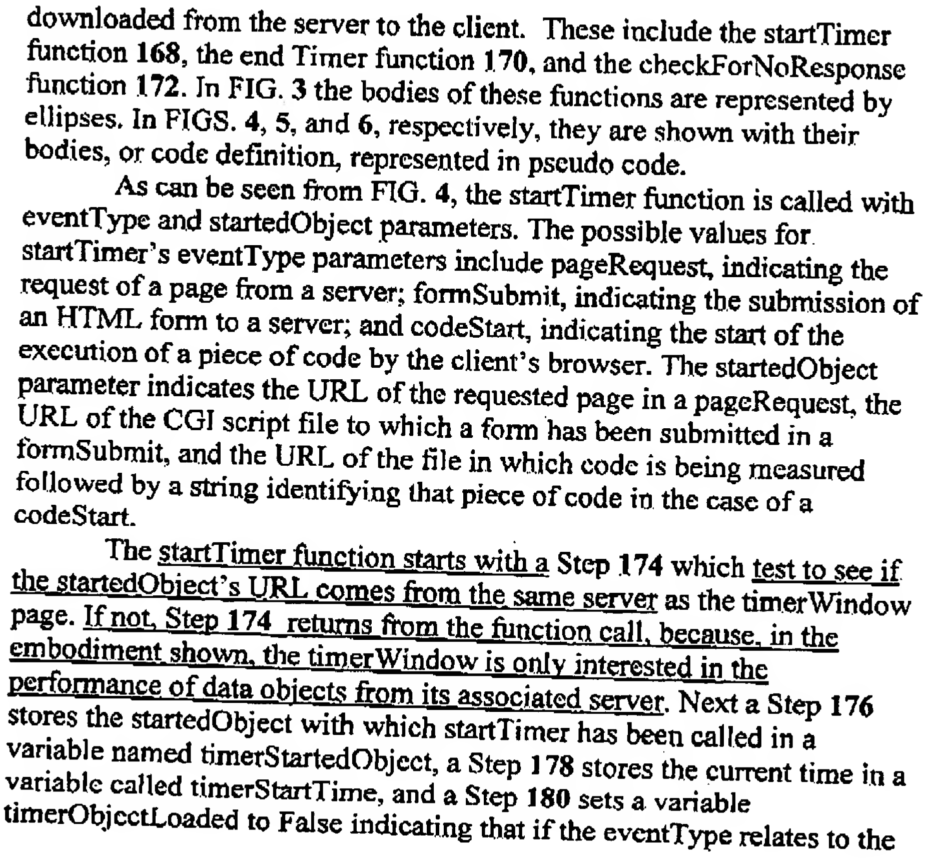
In the Office Action, the Examiner stated the following in regard to Applicants' key/value data structure, which in the original claims was recited as a table in Claim 2 but was not referred to in Claim 1:

> 7. As per claim 2, Killian discloses a key/value table, wherein the key is a cookie in a request header [col 3, lines 40-46] and the value is a time stamp signifying the service time for a request [col 11, lines 1-19].
>
> Office Action dated December 15, page 4.

The above citation to col. 3, lines 40-46 of Killian reads as follows:

> In the preferred embodiments of the invention shown below such performance monitoring instructions include JavaScript contained in downloaded Web documents and cookies sent to client Web browsers in the HTTP headers with which such data objects are sent. In other embodiments other forms of instructions, such as, for example, Java or ActiveX applets, can be used.

The above citation to col. 11, lines 1-19 of Killian, together with the adjacent section col. 11, lines 20-31, reads as follows:



downloaded from the server to the client. These include the startTimer function **168**, the end Timer function **170**, and the checkForNoResponse function **172**. In FIG. **3** the bodies of these functions are represented by ellipses. In FIGS. **4**, **5**, and **6**, respectively, they are shown with their bodies, or code definition, represented in pseudo code.

As can be seen from FIG. **4**, the startTimer function is called with eventType and startedObject parameters. The possible values for startTimer's eventType parameters include pageRequest, indicating the request of a page from a server; formSubmit, indicating the submission of an HTML form to a server; and codeStart, indicating the start of the execution of a piece of code by the client's browser. The startedObject parameter indicates the URL of the requested page in a pageRequest, the URL of the CGI script file to which a form has been submitted in a formSubmit, and the URL of the file in which code is being measured followed by a string identifying that piece of code in the case of a codeStart.

The startTimer function starts with a Step **174** which test to see if the startedObject's URL comes from the same server as the timerWindow page. If not, Step 174 returns from the function call, because, in the embodiment shown, the timerWindow is only interested in the performance of data objects from its associated server. Next a Step **176** stores the startedObject with which startTimer has been called in a variable named timerStartedObject, a Step **178** stores the current time in a variable called timerStartTime, and a Step **180** sets a variable timerObjectLoaded to False indicating that if the eventType relates to the

> downloading of a page, that page has not yet been downloaded. (Emphasis added.)

The above two sections of Killian very clearly do not teach any connection or relationship between cookies or other request identifiers and time values. This lack of association is emphasized by the two citations, which are widely separated from one another in the teachings of Killian. That is, the cited reference to cookies sent to client web browsers is at column 3 of Killian, whereas the cited reference to a startTimer is at column 11 thereof.

Even more importantly, Applicants have found no teaching or suggestion, anywhere in Killian, of keys and service time values being grouped in pairs, wherein each key is an identifier present in a URI request. Neither does Killian show or suggest a key/time value pair being associated with a URI request, nor does it teach creating a table or other data structure of the key/time value pairs. Thus, the Killian reference clearly cannot show the above-stated Feature (1) of Applicants' Claim 1.

Moreover, there is no teaching in Killian of updating each of the service time values, to indicate the time value for completing the responses to respective URI requests. Accordingly, Killian does not show the above Feature (3) of Claim 1.

Finally, Killian does not disclose or suggest the above Feature (5), that is, the step of using the keys and updated time values from the data structure to compute the total time required to respond to all requests associated with an electronic document.

As discussed above, and as taught in the application at page 10, lines 27-32, Applicants' invention is intended to retrieve multiple inline elements pertaining to a web page, wherein the same web server will not be used to deliver all the inline elements. Applicants achieve their intended goal by means of the key/time value data structure recited by Claim 1, in combination with the other recited Claim 1 elements. By contrast, Killian expressly teaches, at col. 11, lines 20-25, that the timer function thereof, discussed at col. 11, lines 1-19, is operable only if each URL "comes from the same server" as the original page. Thus, Killian teaches away from the need for Applicants' invention. Killian thereby teaches away from any motivation for the combination of Claim 1, including the key/time value pairs data structure thereof.

Applicants have considered the teachings of Farber et al. However, none of such teachings, either alone or in any combination with Killian, would overcome the deficiencies of Killian discussed above in regard to Applicants' Claim 1.

As is very well known, references may not be combined under 35 U.S.C. §103 unless the prior art teaches some reason or motivation for making the combination. Moreover, such motivation must come from within the references themselves or from other known prior art, not from Applicants' claims or other teachings. The Killian and Farber references cited by the Examiner are both very complex arrangements that are unrelated to and have no need for one another. The references themselves do not provide any reason or motivation to combine them in order to realize Applicants' Claim 1. In fact, as discussed above, Killian teaches away from Claim 1. Accordingly, Applicants consider that one of skill in art would not be motivated to combine Killian and Farber, as claimed by Applicants, particularly in the absence of a citation to a prior art reference teaching such combination or some motivation therefor.

Independent Claims 7 and 13 recite subject matter similar to subject matter of Claim 1, and are each considered to patentably distinguish over the art for the reasons given in support for Claim 1.

IV. Response to Rejection of Remaining Claims

Claims 2-6 respectively depend from Claim 1, and are each considered to patentably distinguish over the art for the reasons given in support thereof.

Claim 2 is additionally considered to distinguish over the art in reciting that the table of key/time value pairs is created and updated by at least one reverse proxy server associated with the origin server. Neither Killian nor Farber et al, nor any combination thereof, is considered to show or suggest this feature.

Claim 6 is additionally considered to distinguish over the art, particularly in reciting, in the over-all combination of Claim 6, that the controlling monitor sends both sample-on and sample-off commands to the subordinate servers. Claim 6 further distinguishes over the art in reciting the specific tasks that are performed by the subordinate servers, in response to both the sample-on and sample-off commands.

In the Office Action, the Examiner stated the following in regard to Applicants' Claim 6:

> 11. As per claim 6, Farber discloses
>
> the controlling monitor sends a sample-on command to the subordinate servers [i.e. redirect] [col 8, lines 49-53]; in response to the sample-on command, the subordinate servers record service time metrics for request transactions [i.e. log] [col 9, lines 9-13];
>
> the controlling monitor sends a sample-off command to the subordinate servers; in response to the sample-off command, the subordinate servers send their respective service time records to the controlling monitor [i.e. collect] [col19, lines 36-46]; and
>
> the controlling monitor analyzes and reorganizes the service time records from the subordinate servers into a single record [i.e. merge] [col19, lines 28-33].
>
> Office Action dated December 15, page 5.

In the above statement, sections of Farber at col. 8, lines 49-53, col. 9, lines 9-13 and col. 19, lines 36-46 are cited in regard to features of Claim 6 that are associated with the sample-on and sample-off commands. These sections of Farber read as follows:

> This URL is interpreted when received by the repeater.
>
> The reflector then sends (B3-3) a REDIRECT reply containing this new URL to the requesting client. The HTTP REDIRECT command allows the reflector to send the browser a single URL to retry the request.
>
> B6. Whether the request is reflected or handled locally, details about the transaction, such as the current time, the address of the requester, the URL requested, and the type of response generated, are written by the reflector to a local log file.
>
> On a periodic basis, the master repeater (or its delegate) collects logs from each repeater. The logs collected are merged, sorted by reflector identifier and time stamp, and stored in a dated file on a per-reflector basis. The merged log for a given reflector represents the activity of all repeaters on behalf of that reflector. On a periodic basis, as configured by

> the reflector operator, a reflector contacts the master repeater to request its merged logs. It downloads these and merges them with its locally maintained logs, sorting by timestamp. The result is a merged log that represents all activity on behalf of repeaters and the given reflector.

Farber et al, at col. 8, lines 49-53, discloses a reflector sending a reply containing a URL to a requesting client, or to a browser. Applicants are unable to find any teaching in this citation that suggests a controlling monitor that sends a sample-on command, or even any other command, to subordinate servers. It is to be kept in mind that Applicants' Claim 6 depends from Claim 5, which defines both the controlling monitor and subordinate servers of Claim 6 to be reverse proxy servers.

The Farber citation to col. 9, lines 9-13, appears to disclose a reflector writing current time to a local log file. However, such citation does not disclose or suggest subordinate servers recording service time metrics in response to a sample-on command. The Farber citation to col. 19, lines 36-46 discloses a reflector periodically requesting logs from a master repeater. There is no teaching of a controlling monitor sending a sample-off command to subordinate servers, or that the subordinate servers send service time records in response to the sample-off command, as recited by Applicants' Claim 6.

Claims 8-12 and 13-18 depend from Claims 7 and 13, respectively, and are each considered to patentably distinguish over the art for reasons given in support thereof.

Claims 8 and 13 are additionally considered to distinguish over the art for reasons given in support for Claim 2.

Claims 12 and 18 are each additionally considered to distinguish over the art for reasons given in support for Claim 6.

V. Conclusion

It is respectfully urged that the subject application is patentable over the Killian and Farber et al references, and is now in condition for allowance.

The Examiner is invited to call the undersigned at the below-listed telephone number if in the opinion of the Examiner such a telephone conference would expedite or aid the prosecution and examination of this application.

DATE: March 14 , 2005

Respectfully submitted,

James O. Skarst

James O. Skarsten
Reg. No. 28,346
Yee & Associates, P.C.
P.O. Box 802333
Dallas, TX 75380
(972) 385-8777
Attorney for Applicant